AMENDMENTS TO THE CLAIMS

Please enter the following amendments:

1. (Currently Amended) A method of processing data in a single programmable processor, the method comprising:

decoding a single instruction for selectively arranging data, specifying a data selection operand and a first and a second register each having a register width, the single instruction independently specifying the first register and the second register, the first and second registers providing a plurality of data elements each having an elemental width smaller than the register width, the data selection operand comprising a plurality of fields each ~~independently~~ selecting any one of the plurality of data elements and each field having a value not restricted by the other fields included in the data selection operand; and

providing in parallel the data elements selected by the fields to respective predetermined positions in a catenated result, wherein the predetermined positions are in the same order as the fields of the data selection operand.

2. (Previously Presented) The method of claim 1 wherein each field of the data selection operand provides a sufficient number of bits to specify any one of the plurality of data elements.

3. (Original) The method of claim 2 wherein each field of the data selection operand has a width of n bits, wherein the plurality of data elements comprises 2^n data elements.

4. (Original) The method of claim 1 wherein the data selection operand is provided by a register specified by the single instruction.

5. (Original) The method of claim 4 wherein the data selection operand has a width equal to the specified register width.

6. (Original) The method of claim 1 wherein the catenated result is provided to a register.

7. (Original) The method of claim 1 wherein the plurality of data elements has a combined width equal to the width of the first register plus the width of the second register.

8. (Original) The method of claim 1 wherein the instruction further specifies a data element width of the plurality of data elements.

9. (Original) The method of claim 1 wherein each data element has a width of 8 bits.

10. (Original) The method of claim 1 wherein the catenated result has a width of 128 bits.

11. (Original) The method of claim 1 wherein for each field of the data selection operand, a relative location of the field within the data selection operand corresponds to a relative location of the predetermined position within the catenated result.

12. (Previously Presented) The method of claim 1 further comprising:

decoding a second single instruction specifying a third and a fourth register each containing a plurality of floating-point operands;

multiplying the plurality of floating point operands in the third register by the plurality of floating-point operands in the fourth register to produce a plurality of products; and

providing the plurality of products to partitioned fields of a result register as a catenated result.

13. (Canceled)

14. (Currently Amended) A computer-readable storage medium:

having instructions that instruct a computer system to perform operations,

at least some of the instructions including a group element selection instruction for selectively arranging data in a single programmable processor, the group element selection instruction capable of instructing a computer to perform operations comprising:

decoding the group element selection instruction specifying a data selection operand and a first and a second register each having a register width, the group element selection instruction independently specifying the first register and the second register, the first and second registers providing a plurality of data elements each having an elemental width smaller than the register width, the data selection operand comprising a plurality of fields each ~~independently~~ selecting any one of the plurality of data elements and each field having a value not restricted by the other fields included in the data selection operand; and

providing in parallel the data elements selected by the fields to respective predetermined positions in a catenated result, wherein the predetermined positions are in the same order as the fields of the data selection operand.

15. (Previously Presented) The computer-readable storage medium of claim 14 wherein each field of the data selection operand provides a sufficient number of bits to specify any one of the plurality of data elements.

16. (Previously Presented) The computer-readable storage medium of claim 15 wherein each field of the data selection operand has a width of n bits, wherein the plurality of data elements comprises 2^n data elements.

17. (Previously Presented) The computer-readable storage medium of claim 14 wherein the data selection operand is provided by a register specified by the single instruction.

18. (Previously Presented) The computer-readable storage medium of claim 17 wherein the data selection operand has a width equal to the specified register width.

19. (Previously Presented) The computer-readable storage medium of claim 14 wherein the catenated result is provided to a register.

20. (Previously Presented) The computer-readable storage medium of claim 14 wherein the plurality of data elements has a combined width equal to the width of the first register plus the width of the second register.

21. (Previously Presented) The computer-readable storage medium of claim 14 wherein the instruction further specifies a data element width of the plurality of data elements.

22. (Previously Presented) The computer-readable storage medium of claim 14 wherein each data element has a width of 8 bits.

23. (Previously Presented) The computer-readable storage medium of claim 14 wherein the catenated result has a width of 128 bits.

24. (Previously Presented) The computer-readable storage medium of claim 14 wherein for each field of the data selection operand, a relative location of the field within the data selection operand corresponds to a relative location of the predetermined position within the catenated result.

25. (Previously Presented) The computer-readable storage medium of claim 14 wherein at least some of the instructions further include a group floating point multiply instruction for multiplying floating point data in a programmable processor, the group floating point multiply instruction capable of instructing the computer to perform operations comprising:

decoding the group floating point multiply instruction specifying a third and a fourth register each containing a plurality of floating-point operands;

multiplying the plurality of floating point operands in the third register by the plurality of floating-point operands in the fourth register to produce a plurality of products; and

providing the plurality of products to partitioned fields of a result register as a catenated result.

26 – 39. (Canceled)

40. (Currently Amended) A method of processing data in a single programmable processor, the method comprising:

decoding a single instruction specifying a plurality of registers each having a register width, the plurality of registers independently specified by the single instruction and storing a plurality of data elements each having an elemental width smaller than the register width, an index register storing an index vector comprising a plurality of indices stored in partitioned fields of the index register and a destination register;

wherein each index in the index vector comprises a sufficient number of bits to represent a range of possible index values, the range of possible index values including a different index value for each of the plurality of data elements stored in the plurality of registers, allowing the index to select any data element from the plurality of data elements stored in the plurality of registers;

wherein each index in the index vector ~~independently selects one of the data elements from the plurality of data elements stored in the plurality of registers~~ has a value not restricted by the other indices in the index vector; and

providing in parallel the data elements selected by the indices to respective predetermined positions in the destination register, wherein the predetermined positions are in the same order as the indices stored in the partitioned fields of the index register.

41. (Previously Presented) The method set forth in claim 40 wherein the plurality of registers comprises two registers.

42. (Canceled)

43. (Previously Presented) The method set forth in claim 40 wherein the number of indices stored in the index register is equal to the number of predetermined positions in the destination register.

44. (Canceled)

45. (Previously Presented) The method set forth in claim 40 wherein the index vector comprises n equal-sized indices and the destination register comprises n equal-sized predetermined positions.

46. (Previously Presented) The method set forth in claim 45 wherein the index stored in a lowest order set of bits of the index register provides a data element to a lowest order set of bits of the destination register, the index in a second lowest order set of bits of the index register provide a data element to a second lowest order set of bits of the destination register and the index stored in a highest order set of bits of the index register provides a data element to a highest order set of bits of the destination register.

47. (Previously Presented) The method set forth in claim 40 wherein the destination register is a 128-bit register.

48 – 49. (Canceled)

50. (Currently Amended) A method of processing data in a single programmable processor, the method comprising:

decoding a single instruction specifying a first register storing a first plurality of data elements, a second register storing a second plurality of data elements, an index register storing an index vector comprising a plurality of indices stored in partitioned fields of the index register and a destination register;

wherein the single instruction independently specifies the first register and the second register;

wherein each of the first and second registers has a register width, and each of the first and second plurality of data elements has an elemental width smaller than the register width;

wherein each index in the index vector comprises a sufficient number of bits to represent a range of possible index values, the range of possible index values including a different index value for each of the first and second pluralities of data elements stored in the first and second pluralities of registers, allowing the index to select any data element from the first and second pluralities of data elements stored in the first and second pluralities of registers;

wherein each index in the index vector ~~independently selects one of the data elements from the first and second pluralities of data elements stored in the first and second pluralities of registers~~ has a value not restricted by the other indices in the index vector; and

providing in parallel data elements from the first and second pluralities of data elements selected by the indices to respective predetermined positions in the destination register, wherein the predetermined positions are in the same order as the indices stored in the partitioned fields of the index register,

wherein the predetermined positions are contiguous blocks of bits that take up an entire width of the destination register.

51. (Canceled)

52. (Previously Presented) The method set forth in claim 50 wherein the destination register is a 128-bit register.

53. (Canceled)

54. (Currently Amended) A computer-readable storage medium having stored therein a plurality of instructions that cause a single computer processor having registers to perform operations on data elements stored in registers within the processor, the plurality of instructions comprising:

an instruction specifying a plurality of registers each having a register width, the plurality of registers independently specified by the instruction and storing a plurality of data elements each having an elemental width smaller than the register width, an index register storing an index vector comprising a plurality of indices stored in partitioned fields of the index register and a destination register;

wherein each index in the index vector comprises a sufficient number of bits to represent a range of possible index values, the range of possible index values including a different index value for each of the plurality of data elements stored in the plurality of registers, allowing the index to select any data element from the plurality of data elements stored in the plurality of registers;

wherein each index in the index vector ~~independently selects one of the data elements from the plurality of data elements stored in the plurality of registers~~ has a value not restricted by the other indices in the index vector; and

wherein the instruction causes the computer processor to provide in parallel the data elements selected by the indices to respective predetermined positions in the destination register, wherein the predetermined positions are in the same order as the indices stored in the partitioned fields of the index register.

55. (Previously Presented) The computer-readable storage medium set forth in claim 54 wherein the plurality of registers comprises two registers.

56. (Previously Presented) The computer-readable storage medium set forth in claim 54 wherein the plurality of registers comprises two 64-bit registers storing a combined total of sixteen 8-bit data elements.

57. (Previously Presented) The computer-readable storage medium set forth in claim 54 wherein the number of indices stored in the index register is equal to the number of predetermined positions in the destination register.

58. (Canceled)

59. (Previously Presented) The computer-readable storage medium set forth in claim 54 wherein the index vector comprises n equal-sized indices and the destination register comprises n equal-sized predetermined positions.

60. (Previously Presented) The computer-readable storage medium set forth in claim 59 wherein the index stored in a lowest order set of bits of the index register provides a data element to a lowest order set of bits of the destination register, the index in a second lowest order set of bits of the index register provide a data element to a second lowest order set of bits of the destination register and the index stored in a highest order set of bits of the index register provides a data element to a highest order set of bits of the destination register.

61. (Previously Presented) The computer-readable storage medium set forth in claim 54 wherein the destination register is a 128-bit register.

62 – 63. (Canceled)

64. (Currently Amended) A computer-readable medium storage having stored therein a plurality of instructions that cause a single computer processor having registers to perform operations on data elements stored in registers within the processor, the plurality of instructions comprising:

an instruction specifying a first register storing a first plurality of data elements, a second register storing a second plurality of data elements, an index register storing an index vector comprising a plurality of indices stored in partitioned fields of the index register and a destination register;

wherein the instruction independently specifies the first register and the second register;

wherein each of the first and second registers has a register width, and each of the first and second plurality of data elements has an elemental width smaller than the register width;

wherein each index in the index vector comprises a sufficient number of bits to represent a range of possible index values, the range of possible index values including a different index value for each of the first and second pluralities of data elements stored in the first and second pluralities of registers, allowing the index to select any data element from the first and second pluralities of data elements stored in the first and second pluralities of registers;

wherein each index in the index vector ~~independently selects one of the data elements from the first and second pluralities of data elements stored in the first and second pluralities of registers~~ has a value not restricted by the other indices in the index vector; and

wherein the instruction causes the computer processor to provide in parallel data elements from the first and second pluralities of data elements selected by the indices to respective predetermined positions in the destination register, wherein the predetermined positions are in the same order as the indices stored in the partitioned fields of the index register,

wherein the predetermined positions are contiguous blocks of bits that take up an entire width of the destination register.

65. (Canceled)

66. (Previously Presented) The computer-readable storage medium set forth in claim 64 wherein the destination register is a 128-bit register.

67. (Canceled)